Dijkstrův algoritmus

vstup: $G=(V,E,w)$, $u\in V$

výstup: $d(x)\ \forall x\in V$ $d(x)$ délka nejkratší váž. cesty z $u\to x$

$\forall x: d(x) := +\infty$

$z(x) :=$ NEPROBRANÝ

$d(u) := 0$ $p(u) := \emptyset$ heap_insert$(H,(0,u))$ index$(u) := 1$

while $H_1 > 0$:

$(\_, v) :=$ extract_min(H)

$\forall vx \in E$: IF $d(x) > d(v) + w(vx)$:

IF $d(x) = \infty$ insert$(H,(+\infty, x))$ index$[x] := H_1$

$d(x) := d(v) + w(vx)$ $p(x) := v$

heap_decrease$(H, H_1, d(x))$

HALDA s f prvky $T_H^{EXT}, T_H^{UPD}, T_H^{INS} = O(\log f)$

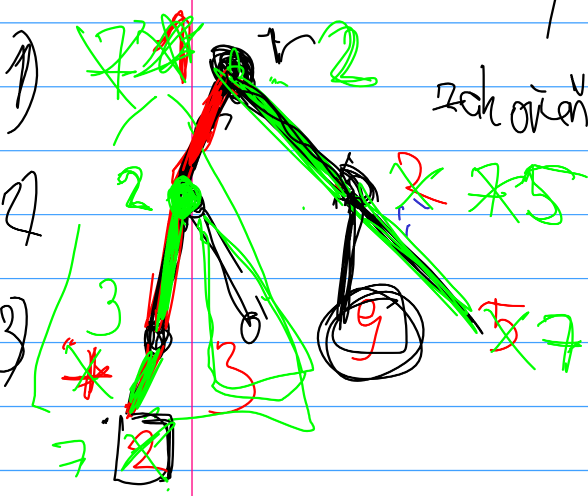
$\Theta(n \cdot T_H^{EXT} + m \cdot T_H^{UPD} + n)$ $\Theta(\log n(n+m))$

$d(x)$ nikdy neroste, $d(x) \geq 0$

PAMĚŤ: $\Theta(n)$ ČAS: $\Theta(n^2 + m)$

$\Theta((m+n)\log n)$ $\Theta(n^2)$

Halda / Heap



zakořen. binární strom s L a P synem u každého vrcholu

$\forall v \in H$: hladina $\equiv$ vzdálenost v od r $+1$

hloubka $H :=$ max hladina $v \in H$

PRAVIDLO 1 (TVAR): všechny hladiny od 1 do (hloubka −1) jsou plně zaplněné & poslední hladina vždy zarovnaná zleva

PRAVIDLO 2 (USPOŘÁDÁNÍ): $\forall\, o, s \in H$ t.ž. o otcem s $\equiv$ s synem o

$$H(o) \leq H(s)$$

REPREZENTACE H má (nejvýše) k vrcholů tak pole o k prvcích H

t.ž. na i-té pozici H uložen hodnota vrcholu j z l-té hladiny pro $i = 2^{\ell} + (j-1)$

$\ell = \lfloor \log_2 i \rfloor$

$$\text{otec}(v_i) = 2^{\ell-1} + \left\lfloor \frac{j-1}{2} \right\rfloor = \left\lfloor \frac{i}{2} \right\rfloor$$

$syn_L(v_i) = 2i$ $\quad$ $syn_R(v_i) = 2i+1$

H = pole hodnot

H_i := počet prvků v H = ℓ

heap_extractmin(H)

```
min := H[1]
H[1] := H[H_i], H_i := H_i - 1
j := 1
while j ≤ H_i :
    if min(H[2j], H[2j+1]) < H[j]:
        if H[2j] < H[2j+1]: j := 2j
        else: j := 2j+1
        swap H[j], H[⌊j/2⌋]
    else: break
return min
```

swap hodnot

swap2 mezi

$O(\log \ell)$

```
heap_insert(H, x)
    H_i := H_i + 1
    H[H_i] := x
    j := H_i
    while (j ≥ 2):
        if H[⌊j/2⌋] > H[j]:
            swap(H[⌊j/2⌋], H[j])
            j := ⌊j/2⌋
        else: break
```

```
H_i := H_i + 1
H[H_i] := +∞
heap_decrease(H, H_i, x)
```

```
heap_decrease(H, j, x):
    H[j] := x
    while j ≥ 2:
        if H[⌊j/2⌋] > H[j]:
            swap(H[⌊j/2⌋], H[j])
            j := ⌊j/2⌋
        else: break
```

MST

Minimální kostra (váženého) grafu

$G = (V, E, w) \quad w: E \to \mathbb{R}$

BÚNO graf neorientovaný

$T \subseteq G$ je kostra (spanning tree) forest

pokud T je strom a $V(T) = V(G)$

pro G souvislý

[Pokud G nesouvislý, tak kostra G
bude sjednocení koster komp. souvislosti G]

kostra T grafu G má $n-k$ hran

k je # komponent G

minimální kostra $G = (V, E, w)$

kostra T t.ž. $\min_{T' \text{ kostra}} w(T') = w(T)$

kde $w(T') := \sum_{f \in E(T')} w(f)$

když T je min. kostra $G=(V,E,w)$

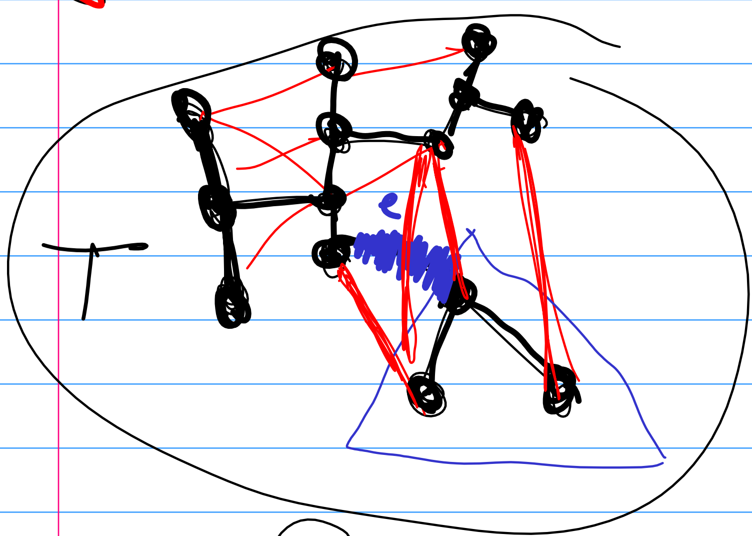
tak T je min. kostra $G'=(V,E,w')$

pro $w'(e) = w(e) + c$

$\forall e \in E$

Algoritmus (Jarník, Prim's algorithm, Dijkstra)

Lemma (ŘEZ): Nechť T je nějaká min kostra G, $e \in E(T)$. Uvážíme-li komponenty C_1 a C_2 v $T-e$ a libovolnou f v G co vede mezi C_1 a C_2



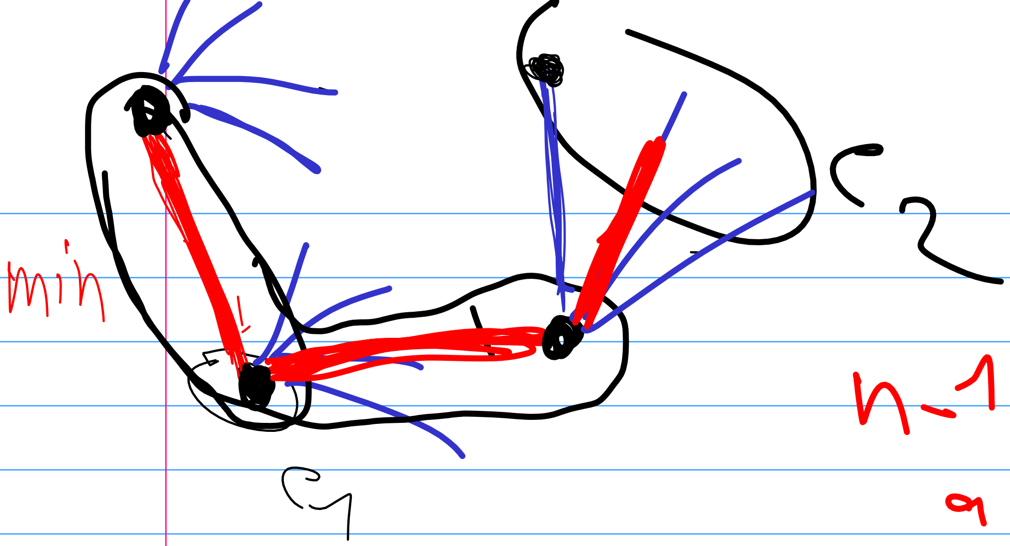
G

tak $w(f) \geq w(e)$

Dk: pro spor $w(f) < w(e)$

tak $T' := T - e + f$

kostra s nižší váhou ↯



$n-1$ hran v acyklickým T $\Rightarrow$ T strom

z lemmatu T je minimální

BÚNO G je souvislý jinak najdi komponenty přes DFS a pusť na komponenty

---

$w : E \to \mathbb{R}$

$w^{ID} := E \to (w(e), \text{identifikátor } e)$

w^{ID} porovnávané lexikograficky

BÚNO

PŘEDPOKLÁDEJME w^{ID} prosté

(V, E, w)

Alg vstup: G $= (V,E,w)$, výstup: T min kostra

$\Theta(n)$

$\forall v$: stav(v) := mimo (aka v C_2)

$h(v) := +\infty$

$p(v) := \emptyset$

$\boxed{H \text{ s sousedy } h(x)}$

v_0 := lib vrchol

$T := \{v_0\}$

stav(v_0) := soused; $h(v_0) := 0$

$\boxed{\text{vrcholy se stav = soused}}$

while $\exists$ soused:

$O(n)$ | u := vrchol s min $h(x)$ & stav(x) = soused

stav(u) = v kostře

$\leq n$ IF $p(u) \neq \emptyset$: přidej vrchol u do T, hranu $\{u, p(u)\}$ do T

pro všechny hrany ux:

$\deg(u)$ if stav(x) = mimo: $\boxed{\text{stav}(x) = \text{soused}}$

if $h(x) > w(ux)$: $\boxed{h(x) := w(ux)}$; $p(x) := u$

Return T

ČAS: $\Theta(n^2)$

PAMĚŤ: $\Theta(n)$ ČAS: $\Theta(m \cdot \log n)$

Věta: Jarníkův alg vrátí MST

Dk: nechť $v_\emptyset \dots v_{n-1}$ je pořadí

$e_1, \dots, e_{n-1}$

vrcholů přidávání do T v ALG

hrany

nechť T' je MST grafu G

$w(T') < w(T)$

$i := \min i$ t.ž. $e_i \notin T'$

$v_i \notin S_\emptyset$

$\subseteq E(T') \cap E(T)$
STROM

$S_\emptyset := \{v_0 \dots v_{i-1}\}$

a $\in S_\emptyset \subseteq \{v_\emptyset \dots v_{i-1}\}$

b je soused v čase (i-1)

potom $h(b) = w(f_0) < w(e_i) = h(a)$

$f_0 :=$ hrana T' spojující S_0 a $V \setminus S_0$

$w(f_0) > w(e_i)$

$T'' := T' - f_0 + e_i$

$w(T'') < w(T')$ $\lightning$ T' min

$w(f_0) < w(e_i)$

$\rightarrow$ TO JE SPOR S

JARNÍK ALG V KROCE i

$\lightning\lightning\lightning$